Tune Computation in ORBIT

Francois Ostiguy
Beam Physics Dept

Tune Computation using FFT

- Look at the transverse position at a specific location for N turns. Peak of FFT power spectrum is the tune.
- Problem I: need many turns to get accurate result
- Problem II: time consuming !
- Advantage: no approximations, works with arbitrary lattice nonlinearities

Tune Computation using Phase Advance Accumulation

- This is is the method used by ORBIT
- Basically a perturbation approach
- Neglects Lattice Functions "distortions" caused by nonlinerarities
- Advantage: Fast !
- Disadvantage: May not always be accurate enough (e.g. Beam-Beam)

ORBIT Phase Advance Accumulation Algorithm

1. Assume LINEAR Lattice Functions (not including linear space charge contribution.)
2. Subtract out dispersive contribution
3. Normalize x,x′ y,y′ coordinates (Floquet Transformation): $x \rightarrow x / \sqrt{\beta}$
4. Compute angle in normalized coordinates
5. Propagate total x,x′,y,y′ using transfer matrix
6. Repeat 2. – 5. and compute angle (phase) change
7. Accumulate phase change.

Some Observations

- When tunes are calculated, all elements that introduce time dependent forces (space charge kicks and RF) should be turned off !
- ORBIT has a flag to notify "nodes" that tune computation is "on".
- The RF cavities currently are unaware of the tune computation flag.
- The dispersive contribution is subtracted out of the horizontal position before applying the transfer map and computing the phase angles.
- *If the RF cavities are not turned off, the subtracted out dispersive contribution is erroneous. This appears to be the source of our problems !*

Issues and Future Work

- Introduction of a large dp/p in the initial distribution does not result in any noticeable tune spread. Consistent with 1st order transfer matrices: lattice chromatic effects are not modeled.
- Turning on space charge does induce the expected tune spread. Consistent with 2D space charge elements, which are non-linear.
- Q: Does ORBIT use "bare lattice" functions or lattice functions corrected to account for (linear) SC ? (appears to be using bare LF)
- We are in the process of debugging/understanding 2nd order elements and acceleration.